

# Aurinkoenergia-areenan kehittäminen

Case: Solar-Arena.com

**Timo Siltala**

**Opinnäytetyö  
Marraskuu 2014**

Tietojenkäsittelyn koulutusohjelma  
Luonnontieteiden ala



JYVÄSKYLÄN AMMATTIKORKEAKOULU  
JAMK UNIVERSITY OF APPLIED SCIENCES



Tekijä(t) SILTALA, Timo	Julkaisun laji Opinnäytetyö	Päivämäärä 21.11.2014
	Sivumäärä 25	Julkaisun kieli Suomi
		Verkkojulkaisulupa myönnetty: ( x )
Työn nimi AURINKOENERGIA-AREENAN KEHITTÄMINEN Case: Solar-Arena.com		
Koulutusohjelma Tietojenkäsittelyn koulutusohjelma		
Työn ohjaaja(t) TUIKKA, Tommi		
Toimeksiantaja(t) Solarcleantec Oy		
<p>Tiivistelmä</p> <p>Opinnäytetyön tarkoituksena oli suunnitella ja toteuttaa asiakkaana toimineelle Solarcleantec Oy:lle verkkopalvelu, jolla tuotaisiin yhteen kaikki kuluttajille aurinkoenergiaa toimittavat osapuolet, ja tuotettaisiin heitä varten palvelu jolla he pystyisivät näyttämään kuluttajille aurinkoenergiaan siirtymisestä syntyvät säästöt samalla omia tuotteitaan ja palveluitaan mainostaen.</p> <p>Järjestelmällä oli asetettu useita vaatimuksia: Globaaleille markkinoille vaadittaisiin monikielisyyttä. Sen tulisi toimia kaiken tyyppisillä selaimilla, mukaan lukien älypuhelimet ja tablettitietokoneet. Sen tulisi olla kuluttajalle tarkoitettuna osiollaan tasapainossa sekä helpon käytettävyyden että uskottavan tarkkuuden suhteen. Lopuksi sen tulisi edistää järjestelmässä asiakkaina olevien yritysten tuotteiden ja palveluiden myyntiä.</p> <p>Koska järjestelmän toiminta ja rakenne oli asiakkaan toimesta jo suurelta osin mietitty, jäi projektin harteille ensisijaisesti sen kehittäminen ja toteuttaminen, mikä jakaantui neljään osaan: Toiminnallisuuden jakaminen tietokantarakenteiksi ja näiden toteuttaminen, pääkäyttäjän sivusto, jolla hallinnoitaisiin järjestelmän asiakkaita ja tietojen ylläpitoa, järjestelmään tuotteitaan esittelevien asiakkaiden kirjaussivusto ja lopulta käyttäjälle suunnattu aurinkoenergialaskuri, joka käyttäisi hyödykseen asiakkaiden syöttämiä laitteita laskennassaan.</p> <p>Projektin skaalan ollessa opinnäytetyön pituuteen nähden suuri, tuloksena syntyi esittelykelpoinen ja pitkälti toiminnallinen järjestelmä, joka pääsi määriteltyihin tavoitteisiinsa suurelta osin. Projektin jatkokehitys ja toiminnallisuuden lisääminen jatkuu vielä opinnäytetyön jälkeenkin.</p>		
Avainsanat (asiasanat) aurinkoenergia, responsiivisuus, PHP, Javascript		
Muut tiedot		



Author(s) SILTALA, Timo	Type of publication Bachelor's thesis	Date 21.11.2014
		Language of publication: Finnish
	Number of pages 25	Permission for web publication: ( x )
Title of publication THE DEVELOPMENT OF A SOLAR ENERGY ARENA Case: Solar-Arena.com		
Degree programme Business Information Systems		
Tutor(s) TUIKKA, Tommi		
Assigned by Solarcleantec Oy		
<p>Abstract</p> <p>The objective was to design and produce a web based service for the customer Solarcleantec Oy, that would bring together all the parties involved in producing consumer solar energy, and create a service that would allow them to show consumers the benefits of solar energy while promoting their products.</p> <p>The service had several conditions to meet: The global market would require a multilingual service. It should work on all types of browsers, including smart phones and tablet computers. The part of the service for consumers should be a balanced mix of ease-of-use and believably accurate results. Finally, it should contribute to the service's paying customers by improving the sales of their products and services.</p> <p>Since the operation and the structure of the service was provided by the customer already largely thought out, was the project's main priority to the development and implementation, which was divided into four parts: The division of functionality into database structure and its implementation, the admin site, which would oversee the customer's accounts and do data maintenance, the clients posting their products and services in the system and finally the solar energy calculator for the end user that would use the client provided products in the calculations.</p> <p>The scale of the project was large in relation to the length of the thesis, which resulted in a presentable and highly functional service that met the given objectives to a large extent. The ongoing development of the project will continue to increase the functionality beyond the scope of the thesis.</p>		
Keywords/tags ( <a href="#">subjects</a> ) web application, solar energy, PHP, Javascript		
Miscellaneous		

TERMIT .....	2
1 JOHDANTO .....	4
2 TUTKIMUSASETELMA.....	5
2.1 Tavoitteet ja rajaukset .....	5
2.2 Tutkimuskysymykset .....	5
2.3 Tutkimusmenetelmät.....	6
3 PERUSTAA WWW-KEHITYKSESTÄ.....	7
3.1 Palvelinpuolen sovellus.....	7
3.2 Tietokanta.....	8
3.3 Käyttäjäpuolen sovellus .....	8
3.4 Responsiivisuus .....	9
4 TOTEUTUS JA TYÖPROSESSI .....	10
4.1 Alkumäärittelyt .....	10
4.2 Tietokantarakenne.....	12
4.3 Pääkäyttäjän käyttöliittymä .....	14
4.4 Asiakkaan käyttöliittymä.....	15
4.5 Käyttäjän käyttöliittymä .....	17
4.6 Käytetyt avoimen koodin lisäosat.....	21
5 POHDINTA .....	23
5.1 Johtopäätökset.....	23
5.2 Tulevaisuus.....	24
6 LÄHTEET.....	26

## TERMIT

<b>ASP.NET</b>	Microsoftin kehittämä web-ohjelmointikehys.
<b>Composer</b>	Järjestelmätason ohjelma riippuvuuksien ylläpitoon PHP:ssa.
<b>Cookie</b>	Suom. Eväste tai Keksi, on sivuston tekemä ja käyttäjän selaimen piilottama pikkutiedosto.
<b>Csv</b>	Comma Separated Values, yksinkertainen taulukkomuotoilu tekstitiedostoon tallennettuna.
<b>HTML</b>	HyperText Markup Language, yksi internetin pääasiallisista merkintäkielistä.
<b>Java</b>	Sun Microsystemsin kehittämä ohjelmistoalusta.
<b>Joomla</b>	Vapaa avoimeen lähdekoodiin perustuva sisällönhallintajärjestelmä.
<b>LAMP</b>	Kokoelma avoimen lähdekoodin ohjelmia jotka yhdessä muodostavat WWW-palvelimen.
<b>Laravel</b>	Vapaa, avoimen lähdekoodin web-ohjelmointikehys PHP-kielille. Pohjautuu Symphonyyn.
<b>PSR Standardit</b>	Kokoelma yhteisesti hyväksytyjä PHP koodauskäytänteitä.
<b>Python</b>	Yksinkertaisen syntaksin omaava ohjelmointikieli.
<b>Ruby on Rails</b>	Ruby-kielillä toteutettu web-ohjelmointikehys.
<b>Symfony</b>	Vapaa, avoimen lähdekoodin web-ohjelmointikehys PHP-kielille.
<b>Utf-8</b>	Yksi Unicode merkistöstandardeista eri kirjoitusmerkeille.

<b>Yii</b>	Avoimen lähdekoodin web-ohjelmointikehys PHP- kielelle.
<b>Zend</b>	Avoimen lähdekoodin web-ohjelmointikehys PHP- kielelle.

# 1 JOHDANTO

Aurinkoenergia on nyt ja tulevaisuudessa voimakkaasti kasvava energiamuoto. Aurinkoenergian käyttöönotossa yksityishenkilöillä on kuitenkin useita haasteita joihin vastauksen etsiminen saattaa vaikuttaa hankalalta. Tehtyään oman tutkimuksensa aiheesta ja tehtyään päätöksen vähentää energiankulutustaan aurinkolämpökeräimellä tai tuottaa sähköä tarpeisiinsa aurinkosähköpaneelilla kuluttajalla on vielä useita kysymyksiä joihin vastauksen saaminen nykyisellään vaatii aika teknistä perehtymistä.

Toimeksiantajalla oli tähän kohtaan ajatus – luodaan järjestelmä joka tuo yhteen aurinkopaneelien valmistajat, jälleenmyyjät, asentajat ja asiakkaat globaalisesti. Aurinkoenergian ala on parhaillaan muutoksen kourissa ja kasvamassa kiihtyvää vauhtia. Lähes jokainen valtio on ottanut tavoitteikseen aurinkoenergian määrän lisäämisen ja laitteiden hinnat ovat kysynnän ja teknologian kehityksen myötä pudonneet kilpailukykyisiksi perinteiselle energiantuotannolle. Toimeksiantajana toimi Solarcleantec Oy, ([Solar-Arena.com](http://Solar-Arena.com)).

## 2 TUTKIMUSASETELMA

### 2.1 Tavoitteet ja rajaukset

Opinnäytetyön tavoitteena oli kehittää toimeksiantajan aikaisemmin määrittelemän mallin pohjalta toimiva responsiivinen www-laskentaohjelma sivustolle [www.solar-arena.com](http://www.solar-arena.com), jolla eri osapuolet pystyvät saamaan tarpeisiinsa vastauksia. Haasteena oli lisäksi tehdä järjestelmästä monikielinen globaaliin markkinaympäristöön.

Tapoja toteuttaa responsiivinen järjestelmä on useita. Järjestelmän olisi voinut toiminnaltaan toteuttaa esimerkiksi Joomla:n kaltaisella sisällönhallintajärjestelmällä, mutta ne nähtiin jo alussa liian kankeiksi ja sovelluksen vaatimukseen nähden liikaa muokkausta vaativiksi ollakseen tehokkaita. Lisäksi ainoastaan avoimen lähdekoodin alustat tulivat kyseeseen ylläpidettävyyden ja kustannusten kannalta.

### 2.2 Tutkimuskysymykset

Tässä työssä keskeiset kysymykset liittyvät responsiiviseen ja monikieliseen järjestelmäkehitykseen.

Mitkä ovat sovelluksen vaatimukset?

- Miten sovellus toimii?
- Millä tekniikoilla saadaan tietoturvaa kohennettua?
- Miten sovelluksen käyttökokemus saadaan käyttäjälle sopivaksi?
- Miten vastataan aiheen vaihteleviin tietämyksen tasoihin eri käyttäjien kohdalla?

Mitkä ovat sopivia toteutusvälineitä sovelluksen vaatimuksille?



- Mitä vaihtoehtoja välineistä on järjestelmän toteutukseen?
- Mitkä toteutusvälineet sopivat toimeksiantajan käyttöön?
- Miten eri relevantit välineet eroavat toisistaan kehitystavoiltaan?

Miten vastaan tietokannan suunnittelun tarpeisiin?

- Mitkä ovat valitun toteutustavan edut muihin tapoihin nähden?
- Vastaako valittu toteutustapa mahdollisiin tulevaisuuden tarpeisiin?

## 2.3 Tutkimusmenetelmät

Opinnäytetyö oli luonteeltaan enimmäkseen kehittämistutkimusta. Toimeksiantajalla oli tarve sovellukselle, jonka pohjaa ja toimintaa oltiin heidän taholtaan suunniteltu jo jonkin aikaa. Heidän näkemyksensä oli tosin enimmäkseen toiminnallinen, joten sain itse miettiä tekniikat mitä käyttäen sovellusta tulisi kehittää. Teoriapohja tulisi koostumaan käytettävien teknologioiden soveltuvuudesta tehtävään ja analysoida mahdollisia vaihtoehtoja. Aikaansaadusta tiedosta saataisiin näin alustava ratkaisusuunnitelma, minkä pohjalta sovellusta voitaisiin alkaa toteuttaa.

## 3 PERUSTAA WWW-KEHITYKSESTÄ

### 3.1 Palvelinpuolen sovellus

#### PHP

PHP on palvelinpuolen komentosarjakieli. Kun verkkoselaimesi hakee web-osoitteen, se tekee palvelupyynnön verkkopalvelimelle. Jos palvelupyyntö koskee PHP-sivua kuten vaikka `http://www.yritys.com/koti.php`, verkkopalvelin herättää PHP jäsentäjän ja sanoo, "Hei! Tässä kohtaa tulee tehdä jotain ennen kuin lähetän tuloksen takaisin tämän käyttäjän verkkoselaimeen." PHP jäsentäjä käy läpi `koti.php`:n sisältämän PHP-koodin ja palauttaa aikaansaamansa tuloksen. Tulos lähetetään takaisin verkkopalvelimelle dokumentin HTML-koodina, mikä vuorostaan palautetaan käyttäjän verkkoselaimelle, joka näyttää sen käyttäjälle. (Meloni 2003, 18.)

Muita verkko-ohjelmoinnissa palvelinpuolella yleisesti käytettyjä kieliä ovat esimerkiksi Java, Python, Ruby on Rails ja ASP.NET. Päätös käyttää PHP:ta sovellukseen johtui pitkälti tekijän edeltävästä osaamisesta, sekä alustan suosiosta ja sitä kautta kehityksen jatkuvuuden takaamisesta. Osana LAMP pinoa, PHP on johtava avoimen lähdekoodin verkkoteknologia ja se vahvistuu jatkuvasti: muiden teknologioiden kuten Composer ja PSR standardien yhteensulautumisen kautta se tuo erillisiä yhteisöjä yhteen. (McGuire 2014)

#### CodeIgniter

CodeIgniter on PHP:lla toteutettu ja PHP:n toiminnallisuutta jatkava sovelluskehys. Kuten kaikki yleisimmät sovelluskehukset, sen tehtävänä on helpottaa sovelluskehittämistä ja tuoda yhteen kirjastoja ja käytänteitä joilla sovelluskehittäminen helpottuu ja yhtenäistyy sekä välttää yleisimmät tietoturvaongelmat.

CodeIgniter käyttää MVC- mallia, mikä on yleinen käytössä oleva ohjelmistoarkkitehtuurityyli. MVC lyhennys tulee sanoista model-view-controller eli malli-näkymä-käsittelijä, mikä viittaa näiden kolmen osa-alueen erottamiseen toisistaan toiminnallisiksi osiksi. MVC- mallissa tekninen arkkitehtuuri jaetaan kolmeen eri osioon eli liiketoimintalogiikasta vastaavaan malliin, esityksestä vastaavaan näkymään ja toiminnasta vastaavaan ohjaimeen. Eli kyseessä on arkkitehtuurimalli, joka soveltuu hyvin graafisille käyttöliittymille. (Vahtolam 2010)

Muita PHP:n päälle rakennettuja sovelluskehyskiä ovat esimerkiksi Zend, Symfony, Laravel ja Yii. Koska eri järjestelmistä saatavat edut suhteessa haittoihin olivat melko tasaisia, oli päätös CodeIgniterin käytöstä riippuvainen yrityksestä löytyvään aiempaan osaamiseen. Tämä helpotti myös mahdollista koodin uudelleenkäyttöä yrityksen järjestelmien välillä.

## **3.2 Tietokanta**

### **MySQL**

Tietokannaksi sovellukselle valikoitui MySQL. MySQL on suosittu web-palveluiden tietokanta. CodeIgniter sisältää hyvät metodit MySQL:n turvalliseen käyttöön, eivätkä sovelluksen tarpeet vaatineet tehokkaampaa kantaa. Lisäksi MySQL:n syntaksi ja erikoisuudet olivat tekijälle ennestään tuttuja. MySQL tarjoaa yleispätevän ratkaisun: Se toimii monilla alustoilla, sen kokonaiskustannukset ovat alhaiset ja se on vakaa. MySQL:n dokumentaatio on erinomainen. (Suehring 2002, 11.)

## **3.3 Käyttäjäpuolen sovellus**

### **JQuery**

JQuery on Javascript kirjasto jonka tarkoituksena on tehdä JavaScriptillä ohjelmoinnista helpompaa ja hauskeempaa. JavaScript kirjasto on monimutkainen JavaScript ohjelma joka sekä yksinkertaistaa vaikeita tehtäviä ja ratkoo eri selainten

yhteensopivuusongelmia. Toisin sanoen, JQuery ratkaisee kaksi suurinta JavaScriptin ongelmaa, monimutkaisuuden ja usean eri selaimen yhteensopivuusongelmat. (McFarland 2011, 4.)

## **Bootstrap**

Bootstrap on JavaScriptillä ja JQueryllä toteutettu avoimen lähdekoodin käyttäjäpuolen sovelluskehys. Siinä missä palvelinpuolen sovelluskehys nopeuttaa, yhtenäistää ja lisää tietoturvaa järjestelmälle palvelimella, tekee käyttäjäpuolen sovelluskehys tulosten näyttämisestä käyttäjän selaimessa helpompaa ja yhtenäistää koodauskäytänteitä. Bootstrap on tukenut responsiivisuutta versiosta 2.0 alkaen.

## **3.4 Responsiivisuus**

Responsiivinen suunnittelu tarkoittaa ensisijaisesti sitä, että sisällön rakenne ja asettelu suunnitellaan niin, että se sopii hyvin kullekin laitteelle ja sen normaalille käyttötavalle. Keskeisintä on tällöin laitteen näytön koko. (Korpela 2012) Kun suunnitellaan responsiiviset verkkosivut, säästetään niin toteutus-, kuin ylläpitokustannuksissakin, verrattuna erillisen työpöytäversion, kännykkäversion ja vaikkapa tablettiversion tekemiseen. (Väätäinen 2013, 34.) Tekemällä responsiiviset sivut eri versioiden sijaan eri alustoille, pystyttiin sivustoa ylläpitämään yhdessä paikassa ja pitämään sivuston ulkoasu helposti samanlaisena alustasta riippumatta.

## 4 TOTEUTUS JA TYÖPROSESSI

Toimeksiantaja oli valmiiksi miettinyt yleisellä tasolla haluamansa sovelluksen perusrakenteen, päänäkömät ja toimintalogiikan. Tästä oli myöskin tehty yleiskuva, jossa näkömät kuvattiin niiden päätoimintojen tasolla. Tämän kuvan perusteella sain melko vapaat kädet suunnitella järjestelmän varsinaisen rakenteen, sen tietokannan ja ulkoasun. Varsinainen sovelluksella tavoiteltu toiminta, eli asiakasta kiinnostavat laskennat tulisi teoriatasolla toteuttamaan yrityksen työntekijöinä toimivat fyysikot, joiden tekemät laskukaavat tulisi saavuttaa järjestelmään yhteensopiviksi.

### 4.1 Alkumäärittelyt

#### Monikielisyys

Järjestelmä tulisi kehittää alusta alkaen käännettäväksi usealle kielelle, ja tämä tulisi ottaa huomioon välineitä valitessa ja toteutuksen yhteydessä. Codelgniterissä, missä tuli mukana kieliluokka, tämä tarkoitti kaiken käyttäjälle näkyvän tekstin siirtämistä omiin muuttujiinsa, jotka oli tallennettu avaimineen kielikonfiguraatitiedostoihin. Tämä vaati muun muassa myös sitä, että järjestelmää tulisi kehittää yhdellä pääkielellä, josta muut kielet olisivat käännöksiä. Pääkieleksi valikoitui englanti josta käänsin itse alkuun suomenkieliset konfiguraatitiedostot, ja josta myöhemmin käännätettiin käännöstoimistolla saksa, espanja ja mandariinikiina.

Kielen vaihdon toteuttamiseen käytettiin cookieita, jotta käyttäjien valitsema kieli säilyisi jatkossakin samalla selaimella helposti, ja että etusivulla ja kirjautumissivulla käytetty kieli vastaisi jo kerran valittua eikä sisäänkirjaantumista tarvitsisi suorittaa ennen kieliasetusten yksilöllistä hakua.

#### Käyttäjätasot

Järjestelmän vaatimusten mukaan järjestelmän käyttäjät jakautuisivat kolmeen päätasoon, joista keskimäinen vielä lisäksi kolmeksi valinnaiseksi alatasokseen.

- Ylimpänä on pääkäyttäjä, joka luo ja ylläpitää asiakkaiden tilejä ja pystyy omasta erillisestä käyttöliittymästään tekemään erinäisiä huoltotoimenpiteitä.
- Keskimäisenä on asiakas, eli palvelusta maksava aurinkosähkö- tai aurinkolämpöpaneelien valmistaja, jälleenmyyjä tai asentaja. Asiakas voi edustaa näistä kolmesta tasosta mitä tahansa yhdistelmää. He pystyvät tekemään yritystään edustavan profiilin ja määrittämään laskurissa käytettävät laitteet sekä alueet joilla he haluavat niitä markkinoida.
- Alimpana on laskurin käyttäjä, joka kirjautunut järjestelmään tekemillään ilmaisilla tunnuksilla ja joka siten voi tehdä ja tallentaa laskuja valitsemillaan edellisen tason asiakkaiden syöttämillä laitteilla. Täten käyttäjä tekee vertailut oikeilla, alueeltaan myynnistä löytyvillä laitteilla.

## **Avoin lähdekoodi**

Avoimen lähdekoodin ratkaisuja ja komponentteja pyrittiin suosimaan siellä missä mahdollista. Tämä vähentäisi tarvetta omien vastaavien ratkaisuiden kehittämiseen ja siten säästäisi aikaa joka vastaavan yhtä hyvän menetelmän kehittämiseen kului. Tästä seurasi luonnollisesti mahdollisia konflikteja ylläpidettävyyteen, mikäli mahdollisesti tulisi vastaan pääasiallisia komponentteja, kuten CodeIgniteriä, JQueryä tai Bootstrapia päivitettäessä. Nämä potentiaaliset versioristiriidat koettiin kuitenkin hyväksyttäväksi riskiksi säästettyyn aikaan nähden.

## **Helppokäyttöisyys**

Järjestelmän käyttäjät tulisivat olemaan tavallisia aurinkoenergiasta kiinnostuneita ihmisiä. Täten järjestelmää suunniteltaessa oli tärkeää ottaa huomioon sivuston ergonomia, kieliasu, käyttökokemus ja muut käyttöä helpottavat tekniikat. Kieli ei

saasi olla liian teknistä, ja missä vastaan tulisi mahdollisesti laskennan kannalta pakollisia konstikkaita termejä, täytyisi ne esimerkiksi erinäisillä avusteilla avata käyttäjälle. Tätä silmällä pitäen pitkin työn edistymistä käytiin katsauksia, jossa aurinkoenergiatekniikasta ymmärtämättömiä henkilöitä päästettiin kokeilemaan ja kommentoimaan järjestelmää, ja näiden kommenttien perusteella käytettävyyttä parannettiin. Kaikesta ei kuitenkaan voitu tinkiä, sillä karsittaessa ominaisuuksia helppokäyttöisyyden varjolla, vaarannettiin laskurin tulosten tarkkuutta. Lopputulos päättyi olemaan kompromissi tarkkuuden ja helppokäyttöisyyden välillä, mikä oli työn tarkoitus alun perinkin.

## Tietoturva

Tietoturvaa pyrittiin kohottamaan valitsemalla välineet ja käytänteet mahdollisimman hyvin etukäteen. Vaikka CodeIgniter onkin nykyään jo hieman ikääntynyt sovelluskehys, on se oletusasetuksia käytettäessä vielä varsin turvallinen, eikä siinä ole paljastunut suuria tietoturva-aukkoja. Tietoturvaa lisää vielä se, että käyttäjien salasanat suolattiin ja kryptattiin vahvasti uusimpia suosituksia käyttäen. Vaikka tietokantaan jostain syystä pystyttäisiin murtautumaan, ovat salasanat silti turvassa. Pitämällä palvelimen päivitykset ja asetukset kunnossa ehkäistään palvelintason hyökkäyksiä.

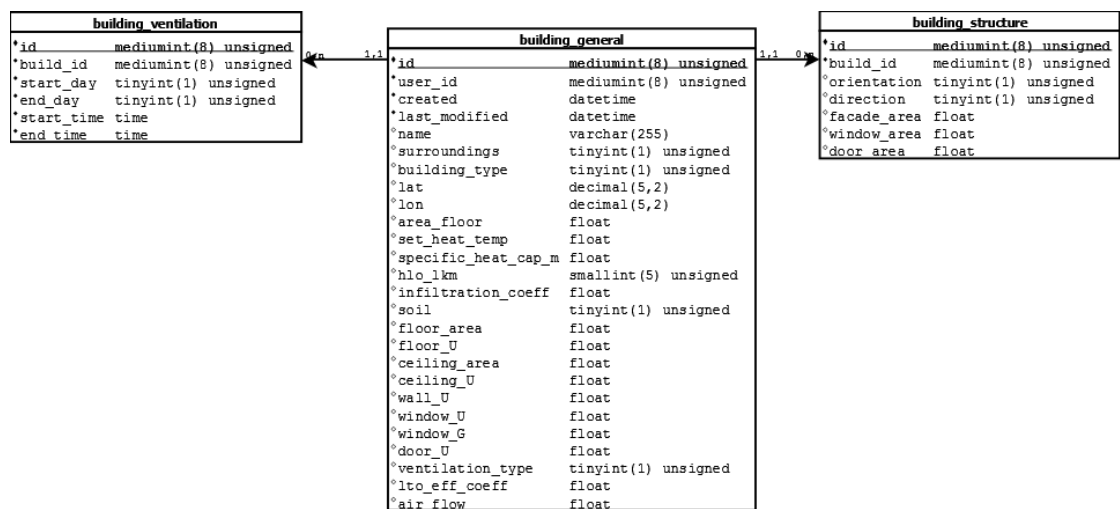
## 4.2 Tietokantarakenne

Tietokantaa suunnittelemaan alettaessa tuli ottaa useita seikkoja. Alussa tietokantarakenne hahmoteltiin paikoilleen miettimällä järjestelmän vaatimuksia.

Pääasiallisia tauluja, joihin suurin osa muista tauluista sitoutui oli useita:

- Järjestelmän kielet
- Käyttäjätyypit
  - Pääkäyttäjä

- Asiakas
- Käyttäjä
- Profiili
- Uutiset
- Tuotekategoriat
- Tuotteet
- Projektit
- Rakennukset



KUVIO 1. Esimerkki tietokannan relaatioista

Laskentaa varten oli projektin toinen työntekijä löytänyt useita laskennassa käytettäviä tauluja jotka tuli kääntää csv-muotoisista tiedostoista MySQL kantaan sopiviksi. Tiedostot olivat suhteellisen kookkaita, esimerkiksi usean vuoden keskiarvotetut säteilymäärät joka kuukaudelle joka koordinaattiparille ( $360 * 180 * 12 = 777600$  yksittäistä arvoa) ja saman verran tietoa keskiarvotetuista kuukausien keskilämmöistä. Lisäksi hänen löytämänsä csv-muotoinen lista maailman kaikista maista ja kaupungeista koordinaatteineen ei ollut täysin yhteensopiva utf-8



kirjoitustyyppin kanssa, mikä aiheutti tietokantaan siirrossa muutaman sadan rivin korjauksen ja siirron käsin.

### 4.3 Pääkäyttäjän käyttöliittymä

Pääkäyttäjän pääasiallinen rooli on luoda ja ylläpitää asiakkaiden tietoja. Lisäksi pääkäyttäjän tilillä voi ylläpitää kielitiedostoja ja luoda ja editoida muiden käyttäjätyyppien kotisivuilla näkyviksi.

## Uutiset

Ylläpitäjä <a href="#">Kotihakemisto</a> <a href="#">Muokkaa Kieliä</a> <a href="#">Muokkaa Käyttäjää</a> <a href="#">Uutiset</a>					
Uusi Artikkel					
#	Otsikko	Valmistajat	Jälleenmyyjät	Asiakkaat	Kirjoitettu
1	18.6.2014 update	✓	—	—	2014-06-18 10:11:24
2	18.6.2014 update	—	—	✓	2014-06-18 08:48:38
3	2.6.2014 update	—	—	✓	2014-06-02 10:26:05
4	2.6.2014 update	✓	—	—	2014-06-02 10:26:05
5	22.5.2014 Update	✓	—	—	2014-05-22 06:54:20
6	22.5.2014 update	—	—	✓	2014-05-22 06:36:04

KUVIO 2. Pääkäyttäjän näkymä uutisiin

Pääkäyttäjän näkymä luotiin oikeastaan jälkiajatuksena, eikä sille alkuvaiheessa ollut edes suurta tarvetta. Osaava ylläpitäjä pystyisi tekemään kaikki muutokset suoraan tietokantaan tai muokkaamaan kielitiedostoja käsin. Mikäli ylläpitoa kuitenkin siirrettäisiin tulevaisuudessa enemmän myynnin puolelle tai tarpeita osittain automatisoituihin tietokannan ylläpitotoimiin alkaisi esiintymään, nähtiin näkymä kuitenkin hyödyllisenä.

## 4.4 Asiakkaan käyttöliittymä

Asiakkaan käyttöliittymä oli ensimmäinen joka järjestelmään luotiin, ja siihen käytettiin eniten aikaa. Koska tyhjällä tietokannalla olisi huono esitellä käyttäjälle yhtään mitään, nähtiin tarpeelliseksi tehdä ensimmäisenä maksavien asiakkaiden käyttöliittymä, joilla he pääsisivät syöttämään tuotteensa ja profiilinsa varsinaisten käyttäjien nähtäville. Lisäksi tekemällä valmistajien näkymä ensimmäisenä vaatimukset järjestelmälle täsmentyivät luonnostaan tekemisen yhteydessä. Asiakastyyppejä määriteltiin alkuun kolme, joista kaikki olivat toistensa kanssa mahdollisia yhdistelmiä. Asiakas saattaisi olla esimerkiksi yhtä aikaa sekä valmistaja ja jälleenmyyjä, tai vaikka jälleenmyyjä ja asentaja.

### Profiili


Profiilinäkymä on yhteinen kaikille kolmelle asiakastyypille. Profiili on heistä käyttäjille näkyvä lyhyt mainososio, jonka käyttäjä näkee tutustuessaan tarkemmin laskurissa näkyvän tuotteen valmistajaan, myyjään tai asentajaan. Profiili on vapaamuotoinen HTML osio, joka voi sisältää melkein mitä tahansa normaalia HTML sisältöä. Se on tarkoitettu esittämään yrityksen logo, lyhyt kuvaus ja linkki yrityksen sivuille tuotteiden yhteydessä, vaikka se ei tähän ole rajoitettukaan. Profiilin voi tuottaa jokaisella haluamallaan järjestelmän ymmärtämällä kielellä, ja yhden kielen voi valita oletuskieleksi, mikäli asiakas selaa sivustoa kielellä mille asiakas ei ole tehnyt profiilia.

## Current Profile

New Profile

English

Solar Business Example



We make solar energy effortless!

Profile In Use

example

Edit Profile

KUVIO 3. Profiiliesimerkki

## Valmistaja

Valmistaja on nimensä mukaan aurinkosähkö- ja aurinkolämpökeräimien valmistaja. Valmistaja on ainoa luokka joka saa syöttää uusia tuotteita tietokantaan ja muokata tai poistaa niitä. Vaikka jälleenmyyjät voivat lisätä tai muokata tuotteisiin omia kuvauksiaan, on valmistaja ainoa asiakasluokka joka voi syöttää tai muokata tuotteen laskentaan vaikuttavia arvoja. Valmistajat ovat siis tavallaan ylin asiakasluokka, jonka syöttämiä tuotteita muut asiakasluokat käyttävät järjestelmässä. Koska suurin osa valmistajista ei myy suoraan loppukäyttäjilleen, oli tämä taso tarpeellinen. Valmistajat ovat käsitteeltään globaaleja, joten heidän tuotteitaan ei rajoiteta maantieteellisesti oletuksena.

## Jälleenmyyjä

Jälleenmyyjät valitsevat valmistajien syöttämien tuotteiden listalta myymänsä tuotteet. Tämän jälkeen he voivat muokata tuotteiden kuvausrivejä mieleisikseen lisäten esimerkiksi uusia kieliä tai linkkejä kuviin tai esitteisiin. He eivät kuitenkaan voi muokata tuotteen laskentaan vaikuttavia arvoja, elleivät ole itse myös tuotteen

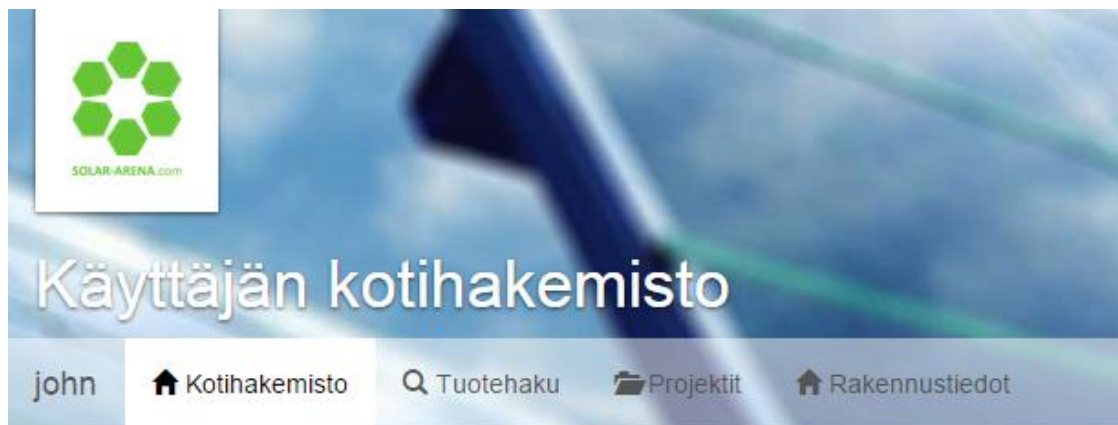
valmistajia. He näkyvät järjestelmässä tuotteiden myyjinä käyttäjien tehdessä kyseisillä tuotteilla laskelmiaan. Toisin kuin valmistajat, jälleenmyyjät voivat määrittää markkina-alueensa millä he näkyvät myyjinä käyttäjille.

## **Asentaja**

Asentajat tarjoavat nimensä mukaan asennuspalveluita. Usein asentajat ovat myös jälleenmyyjiä, myyden sekä paneeleita, tarvikkeita kuten invertterit ja asennustarvikkeet ja näiden asennusta yhtenä pakettina. Käyttäjien näkymässä voidaankin eritellä haluaako asiakas näkyviin laskelmassa käyttämänsä tuotteen jälleenmyyjät, asentajat vai molemmat. Koska asentajat ovat paikallisempia toimijoita kuin valmistajat ja jälleenmyyjät, on heille määritelty myös tarkennettu näkyvyyshmahdollisuus käyttäjille, esimerkiksi jos he tarjoavat asennuspalveluita vain 500km säteellä omasta sijainnistaan, näkyvät he vain käyttäjille jotka ovat määrittäneet sijaintinsa tälle alueelle.

## **4.5 Käyttäjän käyttöliittymä**

Varsinaisten käyttäjien pääsy heitä kiinnostavaan tietoon, eli laskemaan mahdollisia säästöjään eri tasoilla ratkaisulla on tehty suhteellisen helpoksi. Käyttäjien täytyy kirjautua palveluun, koska muuten heidän tekemiään laskuja tai laskujen pohjalla käytettyjä syötettyjä rakennustietoja ei pystytä yksilöivästi tallentamaan. Kirjaututtuaan sisään järjestelmään käyttäjä päätyy alkuun kotinäkömään.



## Uutisia / Päivityksiä

18.6.2014 update

### 18.6.2014 update

#### Features added:

- **Product Search:** Product specs pertaining to product calculation now visible in product info.
- **Calculation:** Recommended tank sizes displayed in an icon next to the tooltip for solar thermal collector systems for each configuration.
- **Calculation:** Savings for photovoltaic modules bar graph can now be produced for all configurations by inputting either the annual energy used (for equal usages each month) or by inputting the individual month usages separately. The numbers used will be saved along with the project.

KUVIO 4. Käyttäjän kotihakemisto

## Asetukset

Käyttäjän tehdessä tili kysytään häneltä sähköpostin, osoitteen ja puhelinnumeron lisäksi jonkin verran pohjatietoja, kuten oletusmaa ja -paikkakunta. Näitä käytetään myöhemmin esimerkiksi rakennustietoja syöttäessä oletuksina. Mikäli käyttäjä haluaa muuttaa asetuksia, hän voi niitä muuttaa täältä jälkikäteen. Lisäksi tästä valikosta voi vaihtaa salasanan ja sivuston ulkoasun teeman.

## **Kotihakemisto**

Kotihakemisto sisältää pääkäyttäjän syöttämät uusimmat uutiset, viimeksi käsitellyt projektit ja pikaohjeen aloitukseen. Kotihakemisto on ensimmäinen paikka minne käyttäjä saapuu kirjaantumisen jälkeen.

## **Tuotehaku**

Tuotehaulla käyttäjä voi selailla järjestelmään syötettyjä tuotteita kategorioittain ja muilla haluamillaan suodatuksilla. Tuotehakua käyttämällä käyttäjä saa yleiskuvaa alueellaan myytävistä laitteiden kirjosta ja niitä myyvistä tahoista.

## **Rakennustiedot**

Rakennustiedot täytyy syöttää aurinkolämpölaskentaa varten. Mikäli asiakas on kiinnostunut vain aurinkosähköstä, ei rakennusta tarvitse syöttää. Rakennuksen syöttäminen on järjestelmän monimutkaisin ja teknisin kohta. Rakennuksen voi syöttää joko käsin mikäli tietää kaikkiin kysyttäviin arvoihin oikeat vastaukset (tätä on pyritty helpottamaan tarjoamalla järkeviä oletusarvoja ja avusteita), tai käyttämällä ohjattua rakennusavustinta joka kyselee käyttäjältä vain ne tiedot mitkä kenen tahansa pitäisi rakennuksesta tietää ja olettaa loput tiedot käyttäen järkeviä, maantieteeseen ja ilmastoon perustuvia oletuksia. Rakennusavustimella tehtyä rakennusta voi luonnollisesti myöhemmin muokata käsinsyötön lomakkeella, mikäli käyttäjä niin haluaa.

Rakennusavustin

Takaisin Rakennukset Menuun

Sijainti ja tyyppi Asukkaat ja ulottuvuudet Energian käyttö

Nimi	Avustettu rakennus	?
Maa	Finland	?
Kaupunki	Äänekoski	?
Ympäristö	Kaupunki	?
Rakennustyyppi	Asunto	?

KUVIO 5. Rakennusavustin

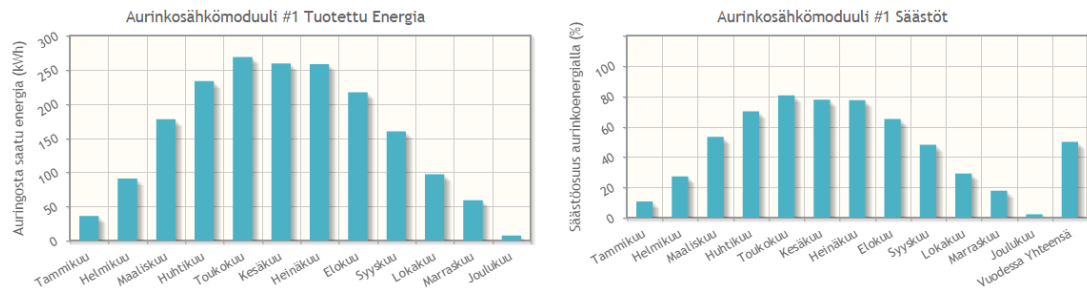
## Projektit

Projektit on käyttäjän kannalta kiinnostavin osio. Täällä käyttäjä luo haluamansa vertailun pohjan, minkä perusteella laskenta suoritetaan. Projektin luotuaan käyttäjä lisää haluamansa määrän aurinkosähkö- tai aurinkolämpöpaneeleita ja määrittää laskussa käyttämänsä rakennuksen (mikäli rakennus on syötetty etukäteen, aurinkosähkölaskuissa pelkkä sijainnin antaminen riittää). Tämän jälkeen käyttäjältä kysytään vielä vähän järjestelmän tietoja ja sitten laskuri palauttaa tulokset sekä lukuina että pylväsdiagrammeina.

Tuloksista ja syöttämistään tiedoista käyttäjä voi päätellä paljon. Jos hän on esimerkiksi laskenut aurinkosähköpaneeleilla saatavaa tuottoa, pystyy hän sähkölaskunsa kokonaissumman kertomalla laskun antamalla säästöprosentilla saamaan vuotuiset säästöt rahassa. Lisäksi, mikäli hänellä on jo sitaatti paneelien hinnasta, pystyy hän täten laskemaan järjestelmän takaisinmaksuajan. Hintoja ei ole järjestelmään toistaiseksi lisätty, koska ne yleensä riippuvat liian monesta seikasta

joita ei sivuilla voida selvittää. Asennuskustannukset ovat lisäksi lähes aina sopimuskohtaisia. Tarjouspyynnön aikaansaanti ja lähetys laskelmasta on kuitenkin sekä asiakkaita että käyttäjiä kiinnostava ominaisuus.

Aurinkosähkömoduulit (tuotetut kilowattitunnit ja säästöprosentti annetusta kokonaiskulutuksesta)													
#	Tam	Hel	Maa	Huh	Tou	Kes	Hei	Elo	Syy	Lok	Mar	Jou	Vuodessa Yhteensä
1	35.8	90.7	177.5	233.2	268.7	259.1	258.2	217.1	159.8	97	59	7.2	1863.4



KUVIO 6. Aurinkosähkölaskelma

## 4.6 Käytetyt avoimen koodin lisäosat

Kaikki projektissa käytetyt avoimen koodin lisäosat olivat liiketoimintaan sopivalla lisenssisopimuksella varustettuja.

<b>PNotify</b>	BootStrappia käyttävä notifiikaatioluokka. Käytetty näyttämään käyttäjille tulevat huomautusviestit.
<b>finderSelect</b>	JQueryllä toteutettu tiedostonkäsittelyluokka. Asiakkaiden liitetiedostojen hallintaan.
<b>CodeIgniter Multi-Upload</b>	CodeIgniterin apukirjasto serveripään tiedostojen vastaanottamiseen.
<b>TinyMCE</b>	Yleisesti käytössä oleva JQueryllä toteutettu HTML editori. Käytetty profiilien ja uutisten syöttöön eri käyttäjänäkymissä.



<b>jquery.cookie</b>	JQueryllä toteutettu cookieiden käsittelyluokka.
<b>Jqplot</b>	JQueryllä toteutettu kuvaajien piirto laskennan tulosten esittelyyn.
<b>jqueryrotate</b>	JQuery kuvien pyörittelyluokka rakennusavustimeen.
<b>tokenfield</b>	JQueryllä toteutettu monivalintakenttä.
<b>Lightbox</b>	JQuery kuvien näyttö omassa ruudussaan.
<b>Bootpag</b>	BootStrap sivutukseen tehty luokka.

## 5 POHDINTA

### 5.1 Johtopäätökset

Alussa suurin haaste oli uusien tekniikoiden opetteleminen ja silti aikatauluissa pysyminen. CodeIgniter ja Bootstrap olivat ennestään minulle tuntemattomia sovelluskehysjä ja niiden opetteluun kuluikin hyvän aikaa, vaikka molempien dokumentaatio olikin toteutettu hyvin. Tämän lisäksi yrityksestä löytyi yksi CodeIgniter osaaja, jonka neuvoihin pystyin tukeutumaan tarvittaessa. Tämä oli itseasiassa sen käyttöön valikoitumisen suurin argumentti. Oppimiskäyrä oli sopivan lineaarinen ja mitä enemmän tutustuin näihin järjestelmiin, sitä enemmän opin arvostamaan niissä tehtyjä ratkaisuja.

Järjestelmä oli jo ajatuksena varsin uniikki. Vaikka erinäisiä aurinkoenergielaskureita löytyy markkinoilta, oli suurin osa niistä laskennallisesti paljon yksinkertaisempia ja yleensä valmistajan ja jälleenmyyjän omiin malleihin sidottuja. Globaaliin laskentaan pystyvää laite- ja kieliriippumatonta ratkaisua ei markkinoilta vielä löydy, ainakaan käyttäjille ilmaista. Voimalaitosten tarpeisiin tarkkaa simulaatiota tuottavia ohjelmia toki löytyy, mutta ne ovat kuluttajan käyttöön kalliita ja vaativat pitkän ja asiantuntevan prosessin antaa vastaava laskelma.

Järjestelmälle löytyisi varmasti käyttöä yhä enemmän uusiutuviin energiamuotoihin panostavassa maailmassa. Yksi syistä lähteä tekemään järjestelmästä oletusarvoisesti globaali oli se, että Suomen markkinat ovat yksistään liian pienet. Näin pohjoisessa ei aurinkoenergia ole vielä kovinkaan varteenotettava tai kuluttajia houkutteleva vaihtoehto, joten vaikka se täälläkin on yleistymään päin, ovat suuremmat markkinat muualla.

Järjestelmää tehdessä keskityttiin ehkä liikaa tarkkuuteen ja ominaisuuksien määrään ja muokattavuuteen käyttökokemuksen kustannuksella. Vaikka järjestelmää muokattiin jatkuvasti helppokäyttöisyyden suuntaan, on se ehkä vieläkin liian hankalasti lähestyttävä kaikille paitsi asiasta jo valmiiksi tietäville ja innostuneille. Jos

käyttäjä vasta harkitsee aurinkoenergian etuja, saattaa hän jättää laskelman teon kesken mikäli vastaan tulee termi joka ei avusteista huolimatta aukene, tai jos kysymysten paljous alkaa tyyppiä kesken lomakkeen täytön.

## 5.2 Tulevaisuus

Käyttöön valitut kielet ja kehykset ovat toistaiseksi maailmanlaajuisesti lajissaan suosituimpien joukossa. Näille löytyy kyllä osaavia jatkajia ja kehitystä pitkälle tulevaisuuteen, joten jatkuvuus ja tietoturvan päivitykset ovat hyvällä mallilla.

Järjestelmää tulee kehittää vielä yksinkertaisemmaksi ja lähestyttävämmäksi. Toistaiseksi järjestelmä vaatii tunnusten luonnin ja sisäänkirjaantumisen, mikä saattaa jo osaltaan käännä osan potentiaalisista käyttäjistä. Laskurista voitaisiin tehdä yksinkertaistetumpi, vielä enemmän olettava (ja siten epätarkempi) versio, jota voisi käyttää ilman sisäänkirjaantumista. Vaikka tämä tarkoittaakin, että laskelmaa ei voisi yksilöivästi tallentaa kantaan, voisi laskelman tekemisestä tehdä niin helppoa, että se ei olisi välttämätöntä. Tai kenties laskelman voisi tallentaa yksilöivällä avaimella joka olisi avoin kaikille linkin tietäville. Täten laskelmia olisi helppo jakaa vaikka sosiaalisessa mediassa.

Ottaen huomioon että järjestelmä alkoi opinnäytetyönä, saattaa sille jatkossa tulla kysymykseen uudelleenkirjoitus paremmalla ja uudemmallalla arkkitehtuurilla. Työtä tehdessä tuli tutustuttua muutamaankin muuhunkin sovelluskehykseen ja niistä löytyi puolia jotka saivat harkitsemaan mahdollisten tulevien projektien tekemistä niillä verrattuna käytössä olleeseen alustaan.

Projekti jatkui ja jatkuu vielä opinnäytetyön jälkeenkin, täysin valmista siitä ei opinnäytetyön aikarajoissa saatu. Osa ominaisuuksista on vielä kesken ja projektin täydellinen suunnanvaihtokaan ei ole poissa kysymyksestä. Ala on nuori ja palvelun bisnesmalli ei ole vielä täysin täsmentynyt joten aika ja asiakkaiden mielenkiinto näyttävät tulevaisuuden suunnan. Tekniseltä puolelta projekti tosin näyttää jo

kyntensä, aurinkoenergiälaskenta järjestelmällä toimii hyvin missä päin maailmaa tahansa.

## 6 LÄHTEET

McFarland, D. 2011. JavaScript & JQuery: The Missing Manual. Sebastopol, CA, USA: O'Reilly Media Inc.

McGuire, J. 2014. Perspectives on the future of PHP. Viitattu 20.11.2014.  
<https://www.acquia.com/blog/perspectives-future-php>

Meloni, J. 2003. PHP Essentials. Boston, MA, USA: Course Technology.

Korpela, J. 2012. Responsiivinen suunnittelu. Viitattu 19.11.2014.  
<http://html5kirja.fi/2012/08/02/responsiivinen-suunnittelu/>

Suehring, S. 2002. MySQL Bible. New York, NY, USA: Wiley Publishing Inc.

Vahtolam. 2010. MVC-Malli, peruskauraa frameworkkien käyttäjille. Viitattu 20.11.2014. <http://vahtolam.wordpress.com/2010/09/23/mvc-malli-peruskauraa-frameworkkien-kayttajille/>

Väätäinen, A. 2013. Responsiivinen Verkkosuunnittelu. Opinnäytetyö. Mikkelin Ammattikorkeakoulu, Tietotekniikan koulutusohjelma. Viitattu 21.11.2014.  
<http://urn.fi/URN:NBN:fi:amk-201302272705>